WATER POLLUTION MONITORING SYSTEM BASED ON MICROCONTROLLER

Compiled as one of the requirements for completing the Strata I Study Programin the Department of Electrical Engineering, Faculty of Engineering

By:
NURUL WAHYU NINGRUM
D400 164 007

ELECTRONIC ENGINEERING STUDY PROGRAM
FACULTY OF ENGINEERING
MUHAMMADIYAH SURAKARTA UNIVERSITY

2021

APPROVAL PAGE

WATER POLLUTION MONITORING SYSTEM BASED ON MICROCONTROLLER

PUBLICATIONS JOURNAL

By:

NURUL WAHYU NINGRUM

D400 164 007

This journal has been checked and approved by:

Supervisor

Heru Supriyono, S.T., M.Sc., Ph.D

NIK. 970

AFIRMATION PAGE

WATER POLLUTION MONITORING SYSTEM BASED ON MICROCONTROLLER

By

NURUL WAHYU NINGRUM

D400164007

Has been tested in front of The Council of Examiners
Faculty of Engineering
Universitas Muhammadiyah Surakarta

The Council of Examiners:

1. **Heru Supriyono, S.T., M.Sc., Ph.D** ( )
   (Head of examiner)
2. **Fajar Suryawan, S.T., M.Eng.Sc., Ph.D** ( )
   (Member I of Examiner)
3. **Pratomo Budi Santoso, S.T., M.T** ( )
   (Member II of Examiner)

The Head of Faculty,

Rois Fatoni, S.T. , M.Sc. , Ph.D.
NIK.892

ACKNOWLEDGMENT

I here by declare that in this publication article there is no work ever submitted to the degree of scholarship in a college and in my knowledge there is also no work or opinion that has been written or published others, except in writing referenced in the manuscript and mentioned in the bibliography.

When there is a proven untruth in my statement above, I will be fully responsibility.

.

Surakarta, 13 August 2021

Author

NURUL WAHYU NINGRUM

D400 164 007

WATER POLLUTION MONITORING SYSTEM BASED ON MICROCONTROLLERMUHAMMADIYAH SURAKARTA UNIVERSITY

Abstrak

Kualitas air menjadi faktor krusial untuk menunjang keperluan mahluk hidup. Faktor yang harus diperhatikan adalah kualitas air. Tingkat kelayakan air yang digunakan bergantung pada tingkat pencemaran air. Penelitian ini dilakukan dengan tujuan untuk mengembangkan sistem pemantau pencemaran air berbasis mikrokontroler, sehingga dapat memantau kualitas air secara realtime dan efisien. Metode penelitian ini adalah rancang bangun yang dapat mengukur parameter pada pencemaran air dengan menggunakan sensor temperature DS18B20, TDS dan turbidity mengggunakan mikrokontroler Arduino UNO yang digunakan untuk mengolah data dan NodeMCU ESP8266 sebagai modul Wifi yang akan mengirim data dari Arduino UNO ke Antares dan ditampilkan pada LCD dan smartphone secara real-time. Berdasarkan pengujian yang telah dilakukan, hasil pengukuran pada sensor suhu memiliki selisih dibawah 1℃ dengan termometer, pada sensor turbidity memiliki hasil pengukuran dengan nilai NTU yang tinggi terdapat pada air keruh yang mana semakin keruh air yang diukur maka semakin tinggi nilai NTU, dan sensor TDS juga mampu melakukan pengukuran yang mana menandakan bahwa parameter sensor sudah bekerja dengan baik dan dapat ditampilkan di LCD dan smartphone dengan jarak jauh. Secara keseluruhan rancangan sistem pemantauanpencemaran air sudah dapat bekerja dengan baik.

Kata Kunci: pemantauan, antares, mikrokontroler, NodeMCU ESP8266, Arduino UNO.

Abstract

Water is an essential part to support the needs of living organisms. One of factors that must be considered is water quality. The appropriate use of water depends on the level of water pollution. This study aims to develop water pollution monitoring system with microcontroller-based, in order to monitor water quality inreal time and efficiently. This study used a design method that can measure parameters on water pollution using temperature sensors DS18B20, TDS and turbidity using Arduino UNO microcontroller which had been used to process data and NodeMCU ESP8266 as a Wifi module that sent data from Arduino UNO to Antares and displayed on the LCD and smartphones in real-time. The results of this study show that: 1. There were difference on the measurement results between temperature sensor and thermometer which was below 1℃,
2. The measurement result of turbidity sensor had a high NTU value in cloudy water. If the measured water were cloudier, it would result a higher NTU value. 3. TDS sensor also able to measure which indicate that the sensor parameters were working properly and can be displayed on the LCD and smartphones remotely. Based on these results, it can be concluded that the performance of water pollution monitoring system were operated well.

Keywords: monitoring, Antares, microcontroller, NodeMCU ESP8266, Arduino UNO.

1. INTRODUCTION

Water is a substance composed of the chemical elements that is vital for life. Creatures that live on thisearth could not separated from their necessity on water. If there was no water on this earth, there wouldbe no life on this earth. About 71 percent of the earth's surface consists of water, and it will be disastrous if air is not available in good and proper conditions in terms of quality and quantity (Warlina,2004). However, the rapid growth of humans has accelerated contamination and aggravate water resources. The contamination status of water could be identified based on parameters: biological, chemical, and physical. Biological parameters include the presence of plankton, bacteria and so on. Chemical parameters include pH, dissolved oxygen, nitrate, nitrite, phosphate, and ammonia. Physical parameters include temperature, turbidity, color and odor (Gierdo et al., 2015). Clean water is highly needed by humans, for daily needs, industrial purposes, farm, agricultural and so on.

Several related studies include research on water turbidity tests based on the internet of things, used NodeMCU ESP8266 as its microcontroller with DFRobot SENO189 sensor to detect turbidity and MQTT Cloud to create a remote monitoring system (Iskandar et al., 2019). The determination ofwater quality (good, normal or bad) based on the IoT-based fuzzy classifier method using PH, TDS and turbidity sensors (Ramadhan et al., 2020).

One of the solutions on monitoring water quality is to identify any changes in water quality parameters on a regular basis. Water quality monitoring helps in evaluating the properties and level ofcontrol towards the pollution conditions (Aziz et al., 2020). Researcher designed a tool to develop a microcontroller-based water pollution monitoring system. This system was able to monitor physical parameters such as temperature, turbidity, and TSD (Total Dissolve Solid) directly through the LCDand smartphone. The data was displayed directly on the LCD and smartphone, and stored in the database for further use.

2. METHOD

This study is a development from previous research, with a concept of water monitoring design in which the turbidity sensor module, DS18B20 temperature sensor and TDS could produced an output voltage. The sensor reading output was processed by Arduino UNO and NodeMCU ESP8266 as microcontroller in digital data that stored and sent over the internet, and the results were directly displayed on the LCD and online on smartphone.

2.1 Water Monitoring System Design

This monitoring system was controlled by microcontroller that consist of Wifi NodeMCU ESP8266 module and Arduino UNO board. The Arduino UNO board was in charge of taking data from the sensor voltage output and controlling the output that would be displayed on the LCD. While the NodeMCU ESP8266 was in charge of sending data to the Antares database. The sensors used for monitoring water pollution are the Turbidity Sensor, TDS Sensor, and DS18B20 Temperature Sensor. The design of water pollution monitoring system is described by diagram in Figure 1.

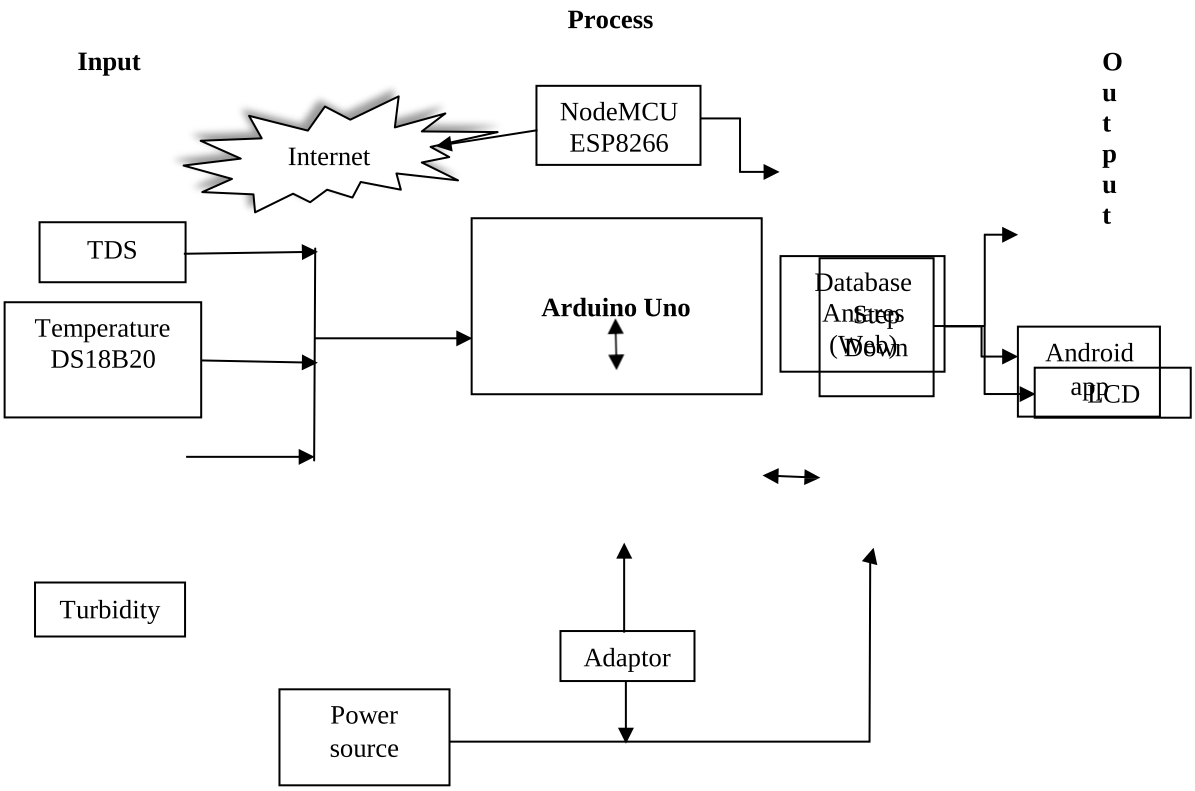


Figure 1. Water Pollution Monitoring SystemHardware and software requirements are presented in Table 1-5.

Table 1. Hardware Device Data

No.	Hardware Unit	Description
1.	Microcontroller	NodeMCU ESP8266
		Arduino Uno
2.	Sensor	Turbidity
		TDS (Total Dissolve Solid)
		Temperature DS18B20
3.	Local display	LCD 12C 16x2
4.	Mobile Wi-fi	Android

Table 2. Specifications of NodeMCU ESP8266

No.	NodeMCU ESP8266	Description
1.	Input voltage	5 VDC
2.	Output voltage	2.5-3.6 VDC
3.	ADC	1 Pin (10 Bit)
4.	GPIO	13
5.	Wi-fi Protocol	802.11 b/g/n/e/i

Table 3. Arduino UNO Specifications

No.	Arduino UNO	Description
1.	Microcontroller	ATmega328
2.	Operating voltage	5V
3.	Input voltage	7-12V
4.	Digital I/O pin	14 (of which 6 provide PWM output)
5.	SRAM	2 KB (ATmega328)
6.	EEPROM	1 KB (ATmega328)

Table 4. Turbidity Specification

No.	Turbidity	Description
1.	Operating current	40Ma (Max)
2.	Operating voltage	3.3-5V
3.	Analog output	0-4.5V
4.	Operating Temperature	5C~90C
5.	Output Method	Analog

Table 5. DS18B20 Temperature Specifications

No.	DS18B20	Description
1.	Power supply	3.0-5.5V
2.	Operating voltage	3.3-5V
3.	Operating Temperature	-55C~+127C
4.	Accuracy over	-10C~+85C: 0.5C

Table 2 shows the general specifications of the NodeMCU ESP8266. This module was an open source IoT platform with the development of a board kit equipped with the ESP8266 module, which integrated GPIO (General Purpose Input/Output), IIC (Inter Integrated Circuit), PWM (pulse width modulation), i-wire and ADC. (Analog to Digital converter). The NodeMCU ESP8266 had analog inputs and 13 GPIO pins (Handson Technology 2016).

Table 3 shows the specifications of the Arduino UNO, with the main component was ATmega328 microcontroller made by Atmel Corporation. Arduino UNO has 14 input/output pins, with 6 pins as PWM outputs, 6 analog inputs, and connected to laptops using USB. According to FeriJuandi (2011:8).

Table 4 shows the specifications of the Turbidity sensor. This turbidity sensor was used to measure the turbidity level of water, with a measuring range of 0-3000 NTU (Nephelometric Turbidity Unit). In this sensor there was a receiver and transmitter of infrared light. Changes in the value of turbidity were caused by the blocking of infrared light as it was a deficiency of light that changed the signal (Ibrahim, 2013).

Table 5 shows the specifications of the DS18B20 temperature sensor. This sensor was used to measure temperature with the DS18B20 chip with a measuring range from -55~125C, which did not require additional external components and included a waterproof sensor.

Figure 1 shows that the display on the LCD/Android app output was affected by the actual changes that are read from the sensor.

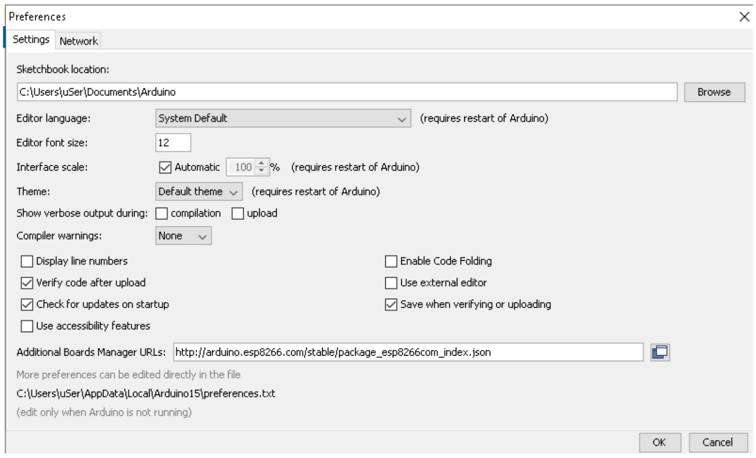
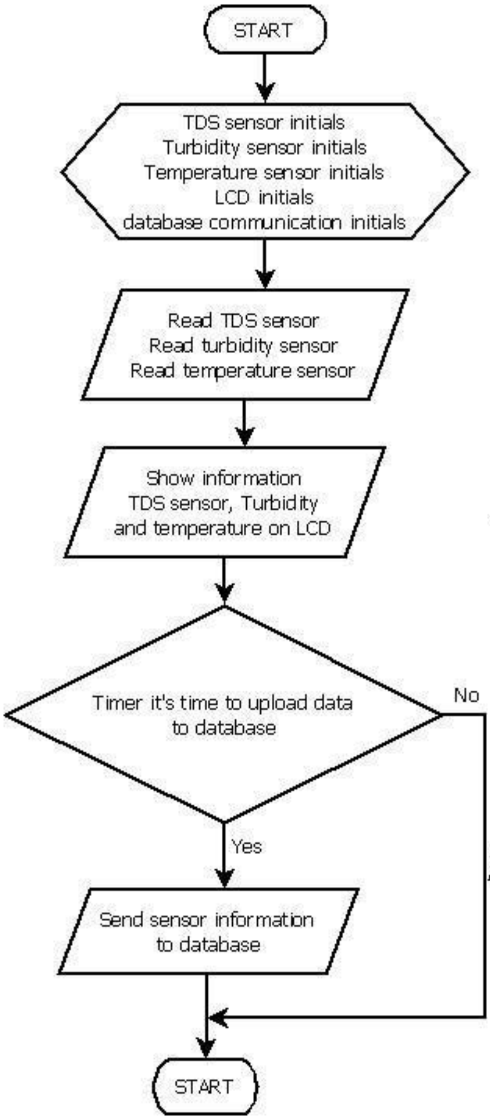


Figure 2. Adding Boardmanager

2.2 Software Design

This program used two Arduino programs that have different purpose, the Arduino Main program had function in reading and processing data that would be uploaded to the Arduino UNO board while the Arduino Nodemcu program had function in sending data from the reading results of the Arduino Main program and then uploaded the program to NodeMCU ESP8266. An additional initialization for sensors, LCD and database communications, which were used as input and output data. Afterwards, the program would read the sensor value and then displayed on the LCD with a delay of 700ms, when arduino program on the timer sent data to the database then the data were uploaded to the database. The initial action of arduino nodemcu program was to add initialization to access the database in the process of reading sensors from the main program, when the timer showed the time for requesting data from the database, the data request would be sent to the database and the database would receive the data information. Then the data was displayed on the user interface.



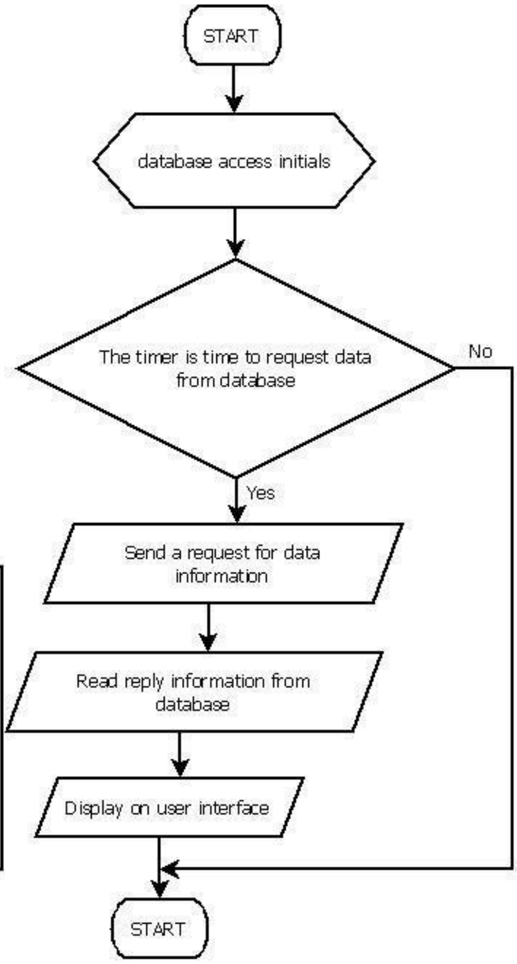


Figure 3. Arduino-NodeMcu Program Flowchart Figure 4. Arduino-NodeMcu Program Flowchart

2.2.1 Arduino IDE Programs.

Before programming the Arduino UNO and NodeMcu ESP8266, it was needed to install the Arduino IDE software and add the ESP8266 board through the preference menu as shown in Figure 2. By adding to the additional board the link http://arduino.esp8266.com/stable/package_esp8266com_index.json in order to access the ESP8266 board that would be used later. Arduino programming, used the C language is used (Iskandar & Zainal2017).

The flow diagram of the programming system is portrayed in Figure 3. The process was started when the sensor input started to read, the results were immediately displayed on LCD and the program continued to run. When the timer had reached the specified time to upload to the database, the sensor value would be sent to the database. In Figure 4, it is shown that data sent to the database would be displayed to the user interface andoid app, when the data in the database was ready to be sent to the user interface. The Arduino UNO and NodeMCU ESP8266 microcontrollers had an important role in the process of each changes on sensor reading, in which Arduino UNO acts as a microcontroller that read the value from the sensor and display it on the LCD display while the NodeMCU ESP88266 microcontroller act as a sender of the sensor value obtained by Arduino UNO to the Antares database and finally displayed to the user interface.

2.2.2 **Arduino IDE Program Instructions**

In application programming of NodeMCU ESP8266 and Arduino UNO, it is needed to use a header through include command, which is portrayed in Figure 5 and Figure 6.

```
#include <SoftwareSerial.h>
#include <AntaresESP8266HTTP.h>
```

Figure 5. Header in Program NodeMCU ESP8266

```
#include "lib_pembacaansensor.h"
#include "lib_server.h"
#include "lib_lcd.h"
```

Figure 6. Header in Program Arduino UNO

Next, determine the output/input on the NodeMCU ESP8266 and Arduino UNO. In the NodeMCU ESP8266 program, an output/input did not require because it was interconnected with Arduino. Input/output program on Arduino Uno is shown in Figure 7.

```
int pinSensorTurbidity = A0;
int pinSensorTDS = A2;
#define ONE_WIRE_BUS A1
```

Figure 7. Input to Arduino UNO Program

After determining the input / output, then continue reading analog input. The following Figure8 is one of the analog input reading commands.

```
float bacaSensorTurbidity(){
  float temporary = 0.0 ;
  temporary = analogRead(pinSensorTurbidity);
  //return temporary;
  float voltage_ = 0.0;
  voltage_ = temporary * (5.0/1024);
  nilai_sensorTurbidity = 100.00-(voltage_/3.82)*100.00;
  return nilai_sensorTurbidity;
```

Figure 8. Input Reading Program on Arduino UNO

Afterwards, displayed the output results as a monitoring tool that can be seen on LCD with thecommand shown in Figure 9.

```
void LCDDisplayUpdate(){

  if( ( millis() - timerUpdateLCD ) > maxTimerUpdateLCD ){

    lcd.clear();

    lcd.setCursor( 0 , 0 );lcd.print("T:");lcd.print(nilai_sensorSuhu, 0);lcd.print("*C");
    lcd.setCursor( 8 , 0 );lcd.print("D:");lcd.print( nilai_sensorTds, 1);lcd.print("ppm");
    lcd.setCursor( 0 , 1 );lcd.print("Turbi:");lcd.print(nilai_sensorTurbidity);lcd.print("NTU");
    timerUpdateLCD = millis();
  }
}
```

Figure 9. Program for LCD Display on Arduino UNO

The NodeMCU ESP8266 was used as remote monitoring, in order to communicate with Antares as the database used, it was needed commands such as accesskey, wifissid, password, projectname and devicename, as shown in Figure 10.

```
#define ACCESSKEY "31421d1c8097d824:7e4f98f240dbdd2c"
#define WIFISSID "LOOK"
#define PASSWORD "N87654321m"

#define projectName "Pemantau"
#define deviceName "PemantauAir"
```

Figure 10. Program for Communication between NodeMCU and Smartphone with Antares

The ESP8266 NodeMCU that was successfully connected to Antares could be recognized bya “request sent” statement at the serial monitor as shown in Figure 11. On the other hand, if the ESP8266 NodeMCU was not connected to Antares, a "connection failed" statement would appeared at the serial monitor, as shown in Figure 12.

```
[ANTARES] Loading root CA certificate success!
[ANTARES] connecting to platform.antares.id
[ANTARES] Server certificate verified
[ANTARES] requesting URL: /~/antares-cse/antares-id/Pemantau/PemantauAir
[ANTARES] request sent
```

Figure 11. Condition of NodeMCU Connected to Antares on Serial Monitor

```
[ANTARES] Loading root CA certificate success!
[ANTARES] connecting to platform.antares.id
[ANTARES] connection failed
```

Figure 12. Condition of NodeMCU Not Connected to Antares on Serial Monitor

2.2.3 Antares Configuration

In addition to the web as a medium of communication (Suryono & Pramusinto, 2016), or fiber optictechnology as a measurement (Omar & MatJafri, 2009), Antares as an IoT platform was used to develop a remote water pollution monitoring system. Data from water pollution monitoring read by NodeMCU ESP8266, next sent to Antares as a storage, after that sent to MIT App Inventor in order toretrieve the latest data received by Antares, and displayed by the smartphone which connected to the internet network.

The initial action to access Antares was to create an account, in order to get an accesskey as a connecting access from NodeMCU ESP8266 to Antares as shown in Figure 13. Next, create a projectname and devicename that would be used as data storage in Antares, shown in Figure 14 .

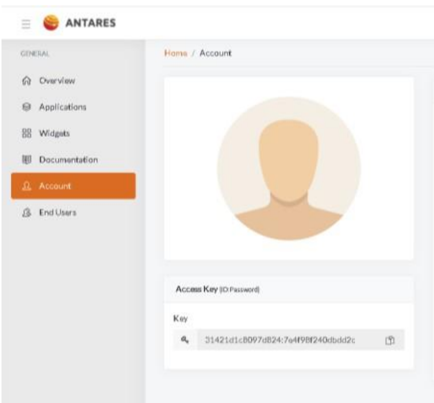


Figure 13. Antares Account Access Key

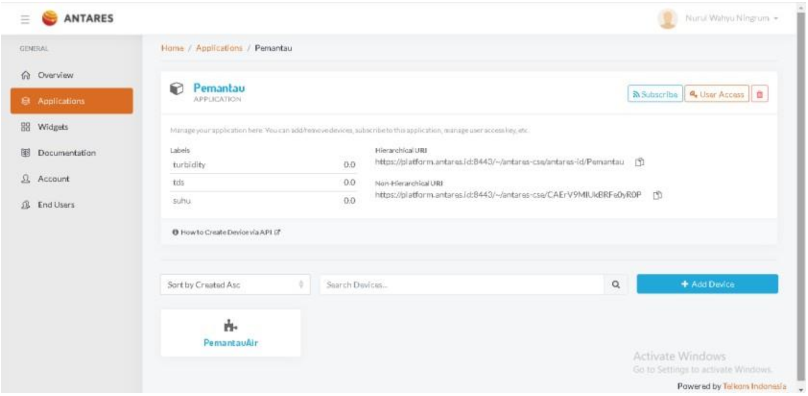


Figure 14. Display Projectname and Devicename

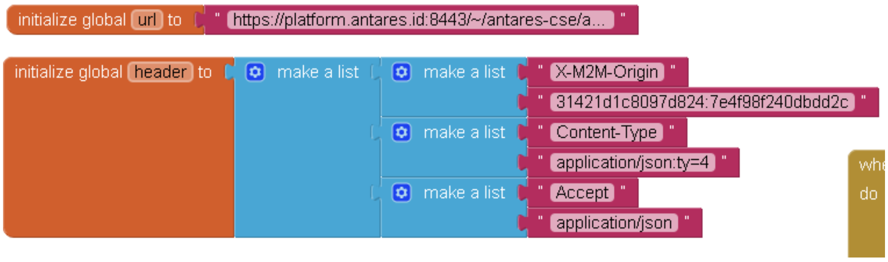


Figure 15. Block from MIT App Inventor

Next, data on the water pollution monitoring display that could be monitored on a smartphone,was obtained from Antares that had been connected to the MIT App Inventor. In order to connect between Antares and MIT App Inventor, a Hierarchical URIwas needed, which comes from the devicename on Antares, is shown in Figure 14, and an access key. The application display can be used with the connected condition to the internet network. Figure 16 shown the display on a smartphone.

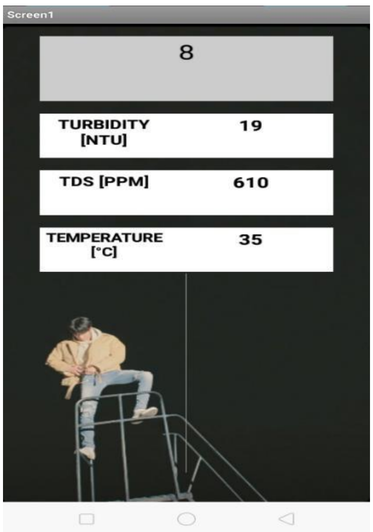


Figure 16. Application on Smartphone Sampling

3. RESULTS AND DISCUSSIONS

A series of tests were carried out as an activity that produced a relationship between the value indicatedby the measuring instrument, or the value represented by the measuring material, with values that hadbeen known and related to the quantity measured under certain conditions.

3.1 Hardware Results

Materials and equipment needed in the manufacture of water pollution monitoring system with microcontroller-based, namely: Wifi NodeMCU ESP8266 module, Arduino UNO, 12 Volt Power Supply/Adapter, DC-DC-STEP-DOWN-LM2596S, Breadboard, Jumper Cable, LCD 12C 16x2, Turbidity Sensor, TDS Sensor and DS18B20 Temperature Sensor. Hardware design was created by using these materials and equipment, as shown in Figure 17.

Figure 17. Hardware Result

3.2 Test Results

3.2.1 Sensor Accuracy Test

The sensor accuracy test was carried out using three water samples: sample 1 was mineral water, sample 2 was river water that had been added with 1 tablespoon of salt and sample 3 was coffee brewed water. It was obtained the measurement results of turbidity, TDS and temperature value.

a. Turbidity Sensor Test

The test was carried out by taking several samples with different levels of turbidity, by inserting theturbidity sensor into different water.

Table 5. Turbidity Sensor Test Data

No.	Sample		
	Mineral water	River water	Coffee
1.	3.24	4.01	39.80
2.	-1.36	4.26	38.52
3.	3.11	4.01	40.43

From these results, it can be concluded that the more turbid the water conditions, the higher the NTU value and vice versa. Changes in the NTU value were caused by the presence of infrared light transmitters and receivers. Iif the water was cloudy, the light on the infrared sensor would be blocked and cause deficiency light and triggered the changes in signal. The standard unit for measuring waterturbidity is NTU (Nephelometeric Turbidity Unit).

b. TDS Sensor Test

In this TDS sensor, the test was performed by adding salt to a sample of
river water.

Table 6. TDS Sensor Test Data

No.	Sample		
	Mineral water	River water	Coffee
1.	773	396	442
2.	766	337	401
3.	767	336	403

The results show that river water that had been mixed with salt had a lower value. In this case it can be concluded that the TDS sensor was not good. Factors that can affect the results of measuring the TDS value was the temperature of liquid, adjustments in the software can also affect the reading accuracy of TDS value in a liquid and the ability of sensor to read the TDS value.

c. DS18B20 Temperature Sensor Testing

Testing on this temperature sensor was performed by comparing measurements from the DS18B20 temperature sensor with a thermometer.

Table 7. DS18B20 Temperature Sensor Test data

Testing	DS18B20 Sensor	Thermometer	Difference
1.	26.88	27	0.12
2.	27.00	28	1
3.	27.06	28	0.94
4.	27.50	28	0.5
5.	27.56	28	0.44
6.	27.56	28	0.44
7.	27.63	28	0.3
8.	50.19	50	0.19
9.	50.44	50	0.44
10.	49.50	50	0.5
11.	48.56	49	0.44
12.	47.63	49	1.37
Average			0.32

Based on experiments results in Table 7, there was difference between temperature sensor witha thermometer which was below 1°C. It can be concluded that the DS18B20 temperature sensor wasperformed well to detect water temperature.

3.2.2 Antares Test with Smartphone

It is needed to ensure that ESP8266 NodeMCU was connected to the internet network, prior testing the data transmission by the ESP8266 NodeMCU to Antares. If it was connected, there would a requestsent statement on the serial monitor which means that the data obtained from the NodeMCU would be sent to the Antares database. In order to test the data that had been sent from the NodeMCU ESP8266 to Antares, an application from the MIT App Inventor was needed as a medium of monitoring water pollution remotely. The results of data received by smartphone from Antares database is shown in Figure 8. Data in the smartphone automatically updated when data from the NodeMCU was sent.

Table 8. Testing Data from Antares Database

Time (WIB)	**Data**		
	Turbidity (NTU)	**Tds (PPM)**	**Temperature (C)**
2021-08-02 03:19:06	3	773	26
2021-08-02 03:20:14	-1	766	27
2021-08-02 03:21:21	3	767	27
2021-08-02 03:22:29	4	336	27
2021-08-02 03:23:36	4	337	27
2021-08-02 03:24:43	4	336	27
2021-08-02 03:25:51	3	331	27
2021-08-02 03:26:58	39	442	50
2021-08-02 03:28:06	38	401	50
2021-08-02 03:29:13	40	403	49
2021-08-02 03:30:21	39	419	48
2021-08-02 03:31:29	39	424	47
2021-08-02 03:32:36	13	427	46

3.2.3 Analysis of Testing Results on Water Pollution Monitoring with Microcontroller-Based

From the experiments that had been carried out by taking data from database, the results show afluctuation measurement values under the same conditions and unable to show a constant output value. Various factors affect the measurement value in non-constant conditions, one of which was due to the sensor itself or the influence of circuit and conditions at the measurement time. The turbidity sensorhad a weakness which is sensitive to sunlight/lights, if there was a difference light in the room, the value would be different.

4. CLOSING

The following conclusions can be drawn from the present study :

a. The test results from the design of a water pollution monitoring system with microcontroller- based show a good performance, although the measurement results were not constant or changeable
b. The display on smartphone as a remote monitor could run well and able to be accessed

anytime

c. Further research is required to evaluate measurements with chemical parameters to produce anaccurate tool.

REFERENCES

Aziz, F. A., Sarosa, M., & Rohadi, E. (2020). Monitoring system water pH rate, turbidity, and temperature of river water. *IOP Conference Series: Materials Science and Engineering*, *732*(1). https://doi.org/10.1088/1757-899X/732/1/012106

Feri, D. (2011). Pengenalan arduino. *Jakarta: Tokobuku. com*.

Gierdo, A., Fisika, J., Sains, F., Diponegoro, U., Gierdo, A., Bangun, R., & Telemetri, S. (2015). RANCANG BANGUN SISTEM TELEMETRI MULTI-CHANNEL UNTUK MONITORING
SUHU DAN pH AIR MENGGUNAKAN JARINGAN WIFI. *Youngster Physics Journal*, *4*(4),257–264.

Handson Technology. (2017). User Manual V1.2: ESP8266 NodeMCU WiFi Devkit. *In Handson Technology Proceedings of the 5th NA International Conference on Industrial Engineering and Operations Management Detroit, Michigan, USA, August* 10 - 14, 2020, (pp. 1– 22).

Ibrahim, S. et al., 2013. International Journal of Science and Engineering (IJSE) Turbidity Measurement Using an Optical Tomography System. *International Journal of Science and Engineering (IJSE)*, 5(October), pp.66–72.

Ibrahim, S. Bin. (2013). Turbidity Measurement Using An Optical Tomography System. *International Journal of Science and Engineering*, *5*(2), 66–72. https://doi.org/10.12777/ijse.5.2.66-72

Iskandar, H. R., Saputra, D. I., & Yuliana, H. (2019). Eksperimental Uji Kekeruhan Air Berbasis Internet of Things Menggunakan Sensor DFRobot SEN0189 dan MQTT Cloud Server. *Jurnal Umj*, *Sigdel 2017*, 1–9.

Omar, A. F. Bin, & MatJafri, M. Z. Bin. (2009). Turbidimeter design and analysis: A review on optical fiber sensors for the measurement of water turbidity. *Sensors*, *9*(10), 8311–8335. https://doi.org/10.3390/s91008311

Ramadhan, M. H., Dewantoro, G., & Setiaji, F. D. (2020). Rancang Bangun Sistem Pakar Pemantau Kualitas Air Berbasis IoT Menggunakan Fuzzy Classifier. *Jurnal Teknik Elektro*, *12*(2), 47–56.https://doi.org/10.15294/jte.v12i2.25351

Suryono, & Pramusinto, K. (2016). Sistem Monitoring Kekeruhan Air Menggunakan Jaringan Wireless Sensor System Berbasis Web. *Youngster Physics Journal*, *5*(4), 203–210.

Warlina, L. (2004). Pencemaran air : sumber, dampak dan penanggulangannya. *Makalah Pribadi*, 1–
26. http://www.rudyct.com/PPS702-ipb/08234/lina_warlina.pdf